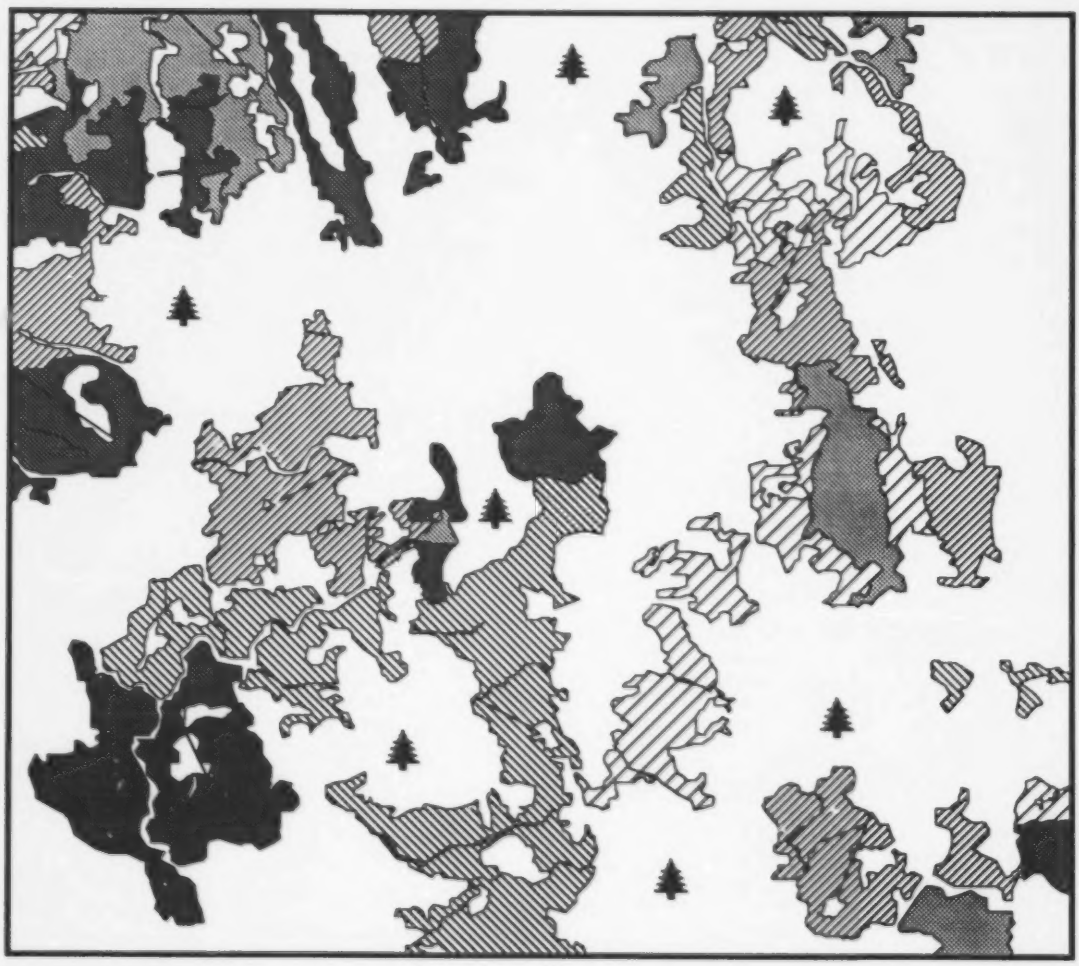
Forestry Canada Forêts Canada

The HSG wood supply model: description and user's manual

T.G.E. Moore and C.G. Lockwood

Information Report PI-X-98
Petawawa National Forestry Institute



91-00353/

PETAWAWA NATIONAL FORESTRY INSTITUTE

In common with the rest of Forestry Canada, the Petawawa National Forestry Institute has as its objective the promotion of better management and wiser use of Canada's forest resource to the economic and social benefit of all Canadians. Objectives of program activities carried out at the Institute support this goal through discovery, development, demonstration, implementation, and transfer of innovations. Because it is a national institute, particular emphasis is placed on problems that transcend regional boundaries or that require special expertise and equipment that cannot be duplicated in Forestry Canada regional establishments. Such research is often performed in close cooperation with staff of the regional centres, provincial forest services, and the forest industry.

Research initiatives and technical services at the Institute encompass five major activities:

FOREST GENETICS AND BIOTECHNOLOGY — Integrates projects in tree genetics, soil microbiology, micropropagation, molecular genetics, meteorology, and seed research. It also includes the client services and seed bank operations of the National Tree Seed Centre, a long-standing program with extensive international affiliations.

FOREST MANAGEMENT SYSTEMS — This program integrates projects in fire, remote sensing, modelling, growth and yield, and forest pest management to provide research and development for the formulation and demonstration of forest management systems.

NATIONAL FOREST RESOURCE STATISTICS — Provides biological, technical, and socioeconomic information on Canada's forest-based resources. The program involves progressive development of databases and establishment of new databases and software in support of policy development in forestry. The Forest Inventory Program collates information on the forest resource at a national level, maintains the Canadian Forest Resources Data System, and prepares the national forest inventory.

COMMUNICATIONS — Integrates activities of the library, public awareness, information, and editing and publications projects. The Institute is visited by more than 20 000 people every year. There is a Visitor Centre for the public, self-guided tours, and an extensive education project. The national repository of all scientific and technical publications of the Forestry Canada and the principal Forestry Canada publications distribution centre are both located at PNFI.

THE RESEARCH FOREST — Besides natural stands manipulated in a variety of ways for silvicultural research, the 100 km^2 Petawawa Forest contains extensive areas of plantations dating back six decades. Research plantations are a source of growth and yield data derived from cultural experiments, and they are becoming valuable for pedigreed genetic materials for micropropagation and molecular genetics studies. The forest also offers opportunities for short- and long-term testing of forest management strategies.

THE HSG WOOD SUPPLY MODEL: DESCRIPTION AND USER'S MANUAL

T.G.E. Moore and C.G. Lockwood

Information Report PI-X-98
Petawawa National Forestry Institute
Forestry Canada
1990

©Minister of Supply and Services Canada 1990
Catalogue No. Fo46-11/98-1990E
ISBN 0-662-18051-8
ISSN 0706-1854
Printed in Canada

Copies of this publication may be obtained free of charge from the following address:

Forestry Canada
Publications Distribution Centre
Petawawa National Forestry Institute
Chalk River, Ontario
K0j 1J0

Telephone: 613-589-2880

A microfiche edition of this publication may be purchased from:

Micromedia Ltd.
Place du Portage
165, Hôtel-de-Ville
Hull, Québec
J8X 3X2

Cette publication est également disponible en français sous le titre
Le modèle HSG de réserve de bois : description et mode d'emploi.

Contents

v Abstract/Résumé

1 Chapter 1. Introduction

2 Chapter 2. Program features and principles of operation
2 Principles of operation
3 Stand update
6 Harvest scheduling
7 Program output

9 Chapter 3. Dataset preparation
9 Stand list
9 Yield look-up tables
10 State table
12 Matching specification
14 Action specification
14 Special characteristics of the state table

15 Treatment priority list
15 Predefined schedule
16 Eligibility database

19 Chapter 4. The Harvest schedule generator
19 Invoking the HSG model
19 Contents of the parameter file
19 Documentation statements
19 TITLE
20 DESC
20 Required input files
20 INVENTORY
20 YIELD
20 STATES
20 Optional input files
20 INDEX
20 CONSTRAINT
20 TREATMENT
20 MANDATORY
20 Optional output files
20 SCHEDULE
21 SNAPSHOT
21 Initialization command
21 BEGIN
21 Run time commands
22 Annotated example of a parameter file

25 Chapter 5. Analyzing and displaying output
25 Summary data
26 SUBSET
27 PS CHART, SUN CHART, TEK CHART, IBM CHART
27 TWOWAY
27 TABLE

28 Literature cited

29 Appendix A.
31 Appendix B

ERRATA

The HSG wood supply model: description and user's manual.
T.G.E. Moore and C.G. Lockwood

Table 3.5 , page 14 should be replaced with the following.

Table 3.5 Format and meaning of the relational operators used in the state table. These relational operators may only be used in combination with the age and Po% fields.

Relational operator	Meaning
=0	match where value equals zero
>0&<31	match where value is greater than zero and less than 31
>30	match where value is greater than 30
*	match any value

Paragraph 3, column 1, page 14: 'distrubance' should read 'disturbance'.

Paragraph 2, column 2, page 20: 'name-number triplet' should read 'type-name-number triplet'.

Figure 4.2 should be added on page 24.

```
treat    natural     0
treat    extensive   1
treat    basic       2
treat    intensive   3
treat    elite       4
dstrb    none        0
dstrb    clearcut    1
```

Figure 4.2 Sample of the INDEX file showing the type-name-number triplets.

Abstract

This document briefly describes a planning system prepared to assist in the design and evaluation of long-range timber harvest schedules. The central component of this planning system is a wood supply model that operates by tracking the development of individual stands through time. The model retains the spatial identity of all stands through all forecasts and thus eliminates forecasting errors caused by aggregation. In areas where geography is an important factor in the design of feasible harvest schedules, the spatially-dependent rules in the harvest scheduling algorithm may substantially improve the feasibility of the computer generated forecasts. The first part of this report describes the features of the model, and the second part provides instructions on how to prepare datasets and operate the software.

Résumé

Le document décrit un système d'organisation lequel a été préparé pour assister à la conception et à l'évaluation des périodes de récolte de bois à long terme. Le composant principal de ce système est un modèle de réserve de bois qui suit le développement de chacun des peuplements dans le temps. Dans le modèle sont inclus les paramètres géographiques de tous les peuplements pour toutes les prévisions. Grâce à ce fait, les erreurs de prévision causées par l'agrégation sont éliminées. Dans les secteurs où l'emplacement géographique joue un rôle important pour déterminer les périodes de récoltes possibles, les règles de l'algorithme qui établissent le plan de récolte et qui dépendent des paramètres géographiques peuvent améliorer substentiellement la possibilité des prévisions générées par l'ordinateur. La première partie du présent rapport décrit la caractéristique du modèle. Les instructions en vue de préparer les bases de données et de faire fonctionner le logiciel se trouvent dans la deuxième partie.

The HSG wood supply model: description and user's manual

Chapter 1 Introduction

This document briefly describes a set of programs prepared to assist in the design and evaluation of long-range timber harvest schedules. These programs have been prepared by the Forest Management Modelling project at the Petawawa National Forestry Institute as part of a cooperative project involving the Iroquois Falls division of Abitibi-Price Inc, the Northern Region of the Ontario Ministry of Natural Resources, and Forestry Canada. The software developed in this project assisted in the preparation of the 1990-1995 timber management plan for the Iroquois Falls Forest Management Agreement.

The central component of this planning system is a wood supply model that operates by tracking the development of individual forest stands through time. The model is a sequential forest inventory projection system similar to FIRFOR (Newnham 1988), FORMAN (Wang et al. 1987), WSM (Baskerville 1982), and WOSFOP (Hall 1978). However, these models disregarded spatial identity when aggregating stands together to form strata. We feel the model described in this paper is a significant improvement over previous models because it retains the spatial identity of all stands described in the forest inventory through all forecasts. This will allow planners to design a location-specific schedule of management activities, and then evaluate the short- and long-term impact of those schedules. In this way site-specific management treatments can be designed and evaluated in light of anticipated forest-level impacts.

Using such a system, the planner can probe questions of a strategic nature, such as "What level of harvest can be sustained from the forest?", or "What type of forest will result from the harvesting operations?", or "How will an increased level of silvicultural treatment alter the sustainable harvest level?". A number of simulation runs can be made where the planner systematically alters program parameters to reflect different management strategies. By comparing output from different simulation runs the planner can estimate the impact that different management strategies would have on the forest. No model can be expected to resolve the ultimate answer to a complicated planning issue, but the planner can use the information from simulation runs to understand the impacts of various management strategies and to help guide the decision on which strategy is most preferable.

As with any quantitative model, the procedures described in this document are based on a numerical interpretation of a real-world problem. The HSG planning model is not intended to provide optimum solutions to timber management planning problems. However, when used in a systematic and well-reasoned manner, quantitative models can provide insight into the nature of "real" problems and can assist in the process of designing and comparing solutions.

The intent of this document is to describe the features of the system and to provide instructions on how to prepare datasets and operate the software. Chapter 2 describes features of the simulation model and the fundamental principles of its operation. Chapter 3 describes the format and contents of datasets required by the model, and provides instructions on how the datasets can be prepared. Commands used to initiate and control the simulations are described in Chapter 4. Chapter 5 describes the format of the output datasets, and the software that can be used to reformat and display the output. Appendix A documents a sample run of the model.

The software described in this document was developed on a SUN microcomputer using the UNIX operating system; this software has also been compiled to run under MSDOS on Intel 80386 computers. Copies of the "C" source code for the programs are available from Forestry Canada. Appendix B describes the computer software and hardware required to run the model. For further information about the software, please write to:

Tom Moore
Petawawa National Forestry Institute
Box 2000
Chalk River, Ontario, Canada
K0J 1J0

Chapter 2 Program features and principles of operation

The central component of this planning system is a wood supply model that operates by tracking the development of individual forest stands through time. Maintaining spatial identity of stands improves the forecasting system in two ways. Firstly, detailed stand-level data can be provided to the growth forecasting procedure. If better growth estimates can be made from stand level data, forecasting errors caused by the aggregation process may be eliminated. Secondly, the harvest allocation is defined on a stand-by-stand basis. It is thus possible to incorporate spatially dependent factors into the harvest scheduling algorithm, and have the model operate in a manner more consistent with the rules applied by operational planners. Because the model explicitly identifies the basis for its forecasts (in terms of the specific stands that are selected for harvest), it is also possible to examine the schedule produced by the model and identify where operational constraints have been violated.

This chapter will briefly describe the principles of operation of the model. The first section will describe the mechanisms used in the model to forecast growth on a stand-level basis, and the application of operational constraints in the harvest scheduling algorithm. Although access to a GIS is not required to run this model, the use of a GIS can significantly reduce the burden of entering and interpreting spatial data. Subsequent sections will describe how a geographic information system (GIS) database can be used to prepare input for the model and to display results from the simulation. The ARC/INFO GIS was used in the prototype system. However, the model was designed to operate with any GIS having a relational database management system.

Principles of operation

The wood supply model operates in a manner similar to other wood supply models such as FIRFOR, FORMAN, WSM, and WOSFOP. An individual-stand forest inventory defines the inital state of the model. Time dependent changes to stand condition are described using yield look-up tables. As the model operates, it sequentially forecasts the changes that would occur in a forest in a series of discrete time steps. The sequence of operations within each time step are: simulate advancing the age of all stands in the forest; update each stand record to simulate stand development (growth or decline); and, simulate the application of harvest and silvicultural operations (Figure 2.1). The simulated time steps (usually five-year periods) are repeated a number of times to forecast the wood supply pattern that would develop over a long-term planning horizon, often 100 years or longer.

The primary database used by the model is a stand list, containing one record for each forest stand in the inventory. The format of the stand list is described in

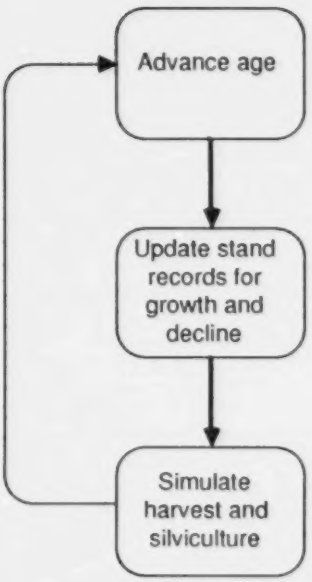


Figure 2.1 Sequence of operations within the simulation model. The cycle of "age, grow, cut" is repeated by the model as far into the future as the planner wishes to explore.

Chapter 3. When the model is run, it first reads the database and stores the stand records in program memory. Each stand has a unique geographic identifier that is a composite of the stand's numeric identifier and its mapsheet location. The relational database management system in the GIS provides a link between the stand records and the spatial descriptions of stand location (Figure 2.2). As long as unique geographic identifiers are used, the GIS can manipulate and display datafiles used or produced by the model.

In the model database, each forest stand is composed of an unlimited number of species sub-components. Each sub-component represent a species, site-class, and age-class type that is present within a stand. Growth and development is forecast independently for each sub-component, thus providing a convenient and flexible means to apply empirical data to describe the development of mixed-species and uneven-aged stands. Each component is linked to a yield look-up table by matching species and site class codes (Figure 2.3). Stand age is the index into the look-up table. The value retrieved from the look-up table is multiplied by the relative stocking factor in order to provide an estimate of current sub-component yield. The yield tables most often describe merchantable volume development of a tree species, but may describe any other forest yield of interest (such as the development of the wildlife habitat characteristics of a stand over time). An unlimited number of sub-components may describe each stand, with each sub-component linked to an inde-

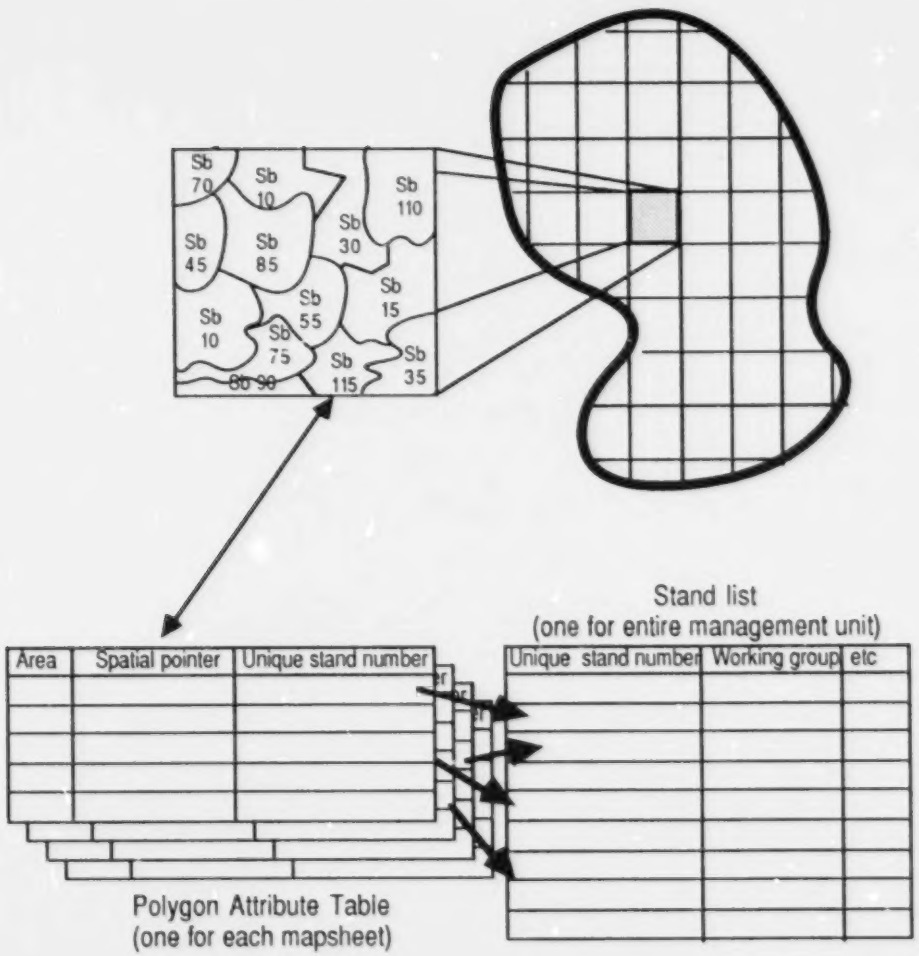


Figure 2.2. The relational database management system in the GIS provides a link between stand records and the spatial description of stand location. The stand identifiers are composed of the map sheet and stand sequence number and uniquely identify each stand in the management unit.

pendent yield curve. By tracking the advance of each sub-component along an appropriate yield curve, the simulation model maintains an up-to-date forecast of sub-component status.

Ageing and increment of time are simulated by advancing a clock that is internal to the model. Unlike other models, the time step by which the clock should be advanced is not fixed. The time increment to be used at each simulation step must be specified as a parameter to the model at each step. As a result, the time step may be short for simulation periods close to the present and may be lengthened for simulation periods far off in the future. While it would be possible to maintain a short time step for all simulation periods, this would require more computations and likely provide an unnecessary amount of detail. By providing a variable time step, planners may examine short-term actions in fine detail, while generalizing about time periods far into the future. For example, a single run of the simulation model may cover the operational planning horizon (first 5-year period) in five 1-year time steps, the management planning horizon (the next 15-year period) in three 5-year time steps, and the strategic planning horizon (several rotations) in 10-year time steps.

Stand update

At the beginning of each simulation step, the model evaluates each stand record and estimates the changes that may have occured since the previous period. There are several steps in the evaluation and update process

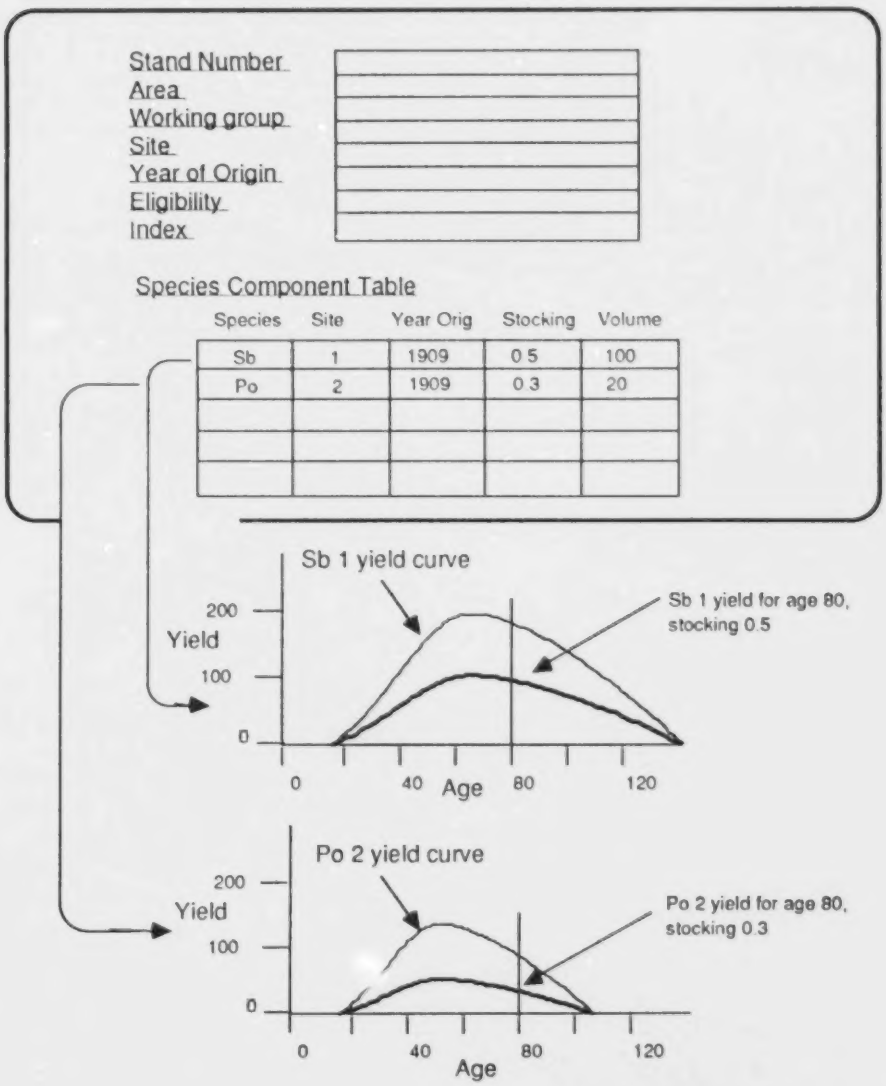


Figure 2.3. Format of the stand list database. An unlimited number of sub-components can occur in each stand, and each sub-component is linked to a yield curve by its species and site-class code. The stocking factor modifies the yield for each sub-component. The format of the stand list required by the model is described in chapter 3. Inventory files must be converted to the required stand list format before the model is run.

(Figure 2.4). After advancing the age of the sub-components, the second step is state-dependent, based on the composition of the stand at the new point in time. State-dependent rules are used to describe changes caused by natural disturbances such as stand breakup and ecological succession, and also from human-caused disturbances harvest and silvicultural treatment. The state-dependent rules are composed of two parts: a matching specification and an action specification. A matching specification identifies the range of stand conditions to which the associated action specification should be applied. The action specification describes changes to be applied to a stand record when a match is found. For example, the breakup of an old-growth black spruce stand and replacement with advance regeneration could be represented with a rule that matches to the anticipated age of breakup, followed by an action specification that resets the stand age to 40-years-old (Figure 2.5).

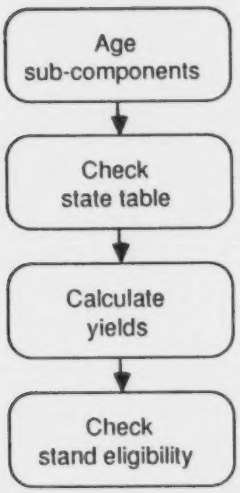


Figure 2.4. Steps in the stand update process. Each stand is evaluated by this process at each iteration of the simulation model.

Current yields are determined in the third step of the update process. This procedure was inspired by the mixed-species yield curve procedure developed by the Wood Supply - Growth and Yield Technical Committee of the New Brunswick Forest Research Advisory Council (NBFRAC 1986). Yields are calculated independently for each stand component and yield look-up tables are used to express time-dependent changes to the stand components. The component species code and site class determine which look-up table to use, and specific values are interpolated for the component age. The resulting value is multiplied by the component stocking factor and is stored in the component volume field (Figure 2.3). The advantage of this approach is that separate yield curves are used to forecast the development of individual species components within a stand (e.g., in a mixed species stand, a spruce curve is used for a spruce component, a poplar curve is used for a poplar component, etc.) This approach can be extended to include product classes as well (e.g., to distinguish spruce fibre from spruce lumber). There is no limit to the number of components that may be defined within a stand.

The final step in the update process checks each stand against an eligibility database. The eligibility database contains records describing time periods when stands may be ineligible for harvest (for example, blocks of stands may be ineligible due to lack of road access, entire working groups or stand types may be restricted due to management guidelines, or single stands may be

Stand breakup rule:

```
Wg Site Age   Po% dist  treat     Spl Sitel Agel Stkgl
bS G    >140  *   none  natural / bS  G     40.  1.0
```

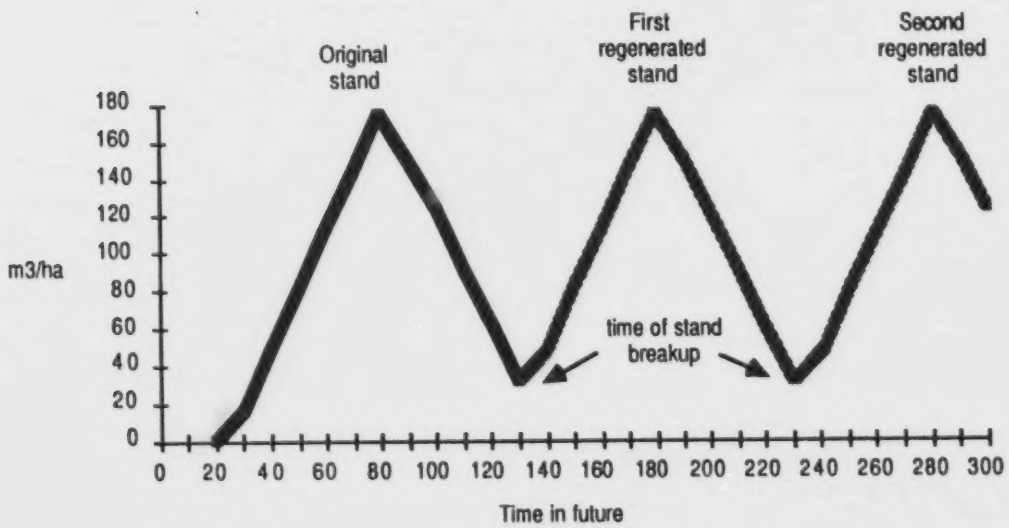


Figure 2.5. Simulation of stand breakup and replacement with advanced regeneration. In this example, the rule in the state table specifies that 140-year-old black spruce stands on site class "G" breakup and are replaced with fully stocked, 40-year-old black spruce regeneration. The series of breakup and regeneration can continue indefinitely in the simulation model.

reserved as wildlife habitats). A detailed description of the eligibility mechanism is provided in Chapter 3. Within this model, the eligibility database is the means by which location- and time-specific constraints are specified to management activities. This mechanism is strictly deterministic: the eligibility database must be defined by the planner before the model is run.

Harvest scheduling

The simulation model allows control over the harvest scheduling process in two steps. First, individual stand harvest treatments may be specified by the planners in an arbitrary, ad-hoc manner (such as harvesting all stands within a primary road access corridor). Secondly, harvest treatments are selected automatically by the model according to a user-specified set of criteria. These steps may be used seperately or in combination.

The harvest scheduling process begins by applying the mandatory harvest schedule. The mandatory harvest schedule is an arbitrary list of stands to be harvested, sorted by the year in which the harvest is to take place. Each mandatory harvest record contains the unique stand identifier, the type of treatment to be performed, the year in which the stand is to be treated, and other descriptive information. The harvest scheduling model reads in the records that apply to the current operating period, and schedules those stands for harvest. Yields from the selected stands are calculated and tallied by species, site-class, and age. The stand records are marked as harvested and schedule records are written to the schedule datafile. The mandatory schedule file used as input has the same file format as the output schedule file. It is thus easy to produce a schedule file with one run of the model, arbitrarily resort and edit the schedule to comply with operational constraints, feed the modified schedule back into the model, and observe the forest-level impacts caused by stand-level changes.

After mandatory harvesting is carried out, the model employs an algorithm to schedule stands for harvest in order to satisfy a user-defined harvest quota. If the quota in the current simulation time step has already been met by the mandatory activities, processing continues with the next step. Similar to previous simulation models, this model employs a simple set of criteria to select stands for harvest treatment. The criteria define a numerical index used to rank the suitability of each stand for harvest. At each time-step of the model, each stand is evaluated with respect to the criteria and then the entire stand list is sorted based on the numerical index.

After sorting the stand list, the model selects the stand at the top of the list as a candidate for harvest. The volume that would be generated as a result of harvesting the stand is computed, a record describing the harvest is written to the schedule file, and the stand record updated to indicate the newly harvested state. The algorithm tallies the cumulative volume that has been scheduled for harvest in the simulation period. If the cumulative harvest volume is less than the user-specified quota for the period, the algorithm will repeat and select the next stand on the list for harvest.

Multiple harvest quotas and selection criteria may be specified within a single simulation period. The model operates by tallying each harvest quota simultaneously, and by applying each of the selection criteria sequentially. Harvest quotas may be specified for the yield of any sub-component or combination of sub-components. For example, a quota specifying the harvest of 100 000 m^3 of black spruce, white spruce or balsam fir would be satisfied when the total amount of harvest of those species, in any proportion, exceeds 100 000 m^3. An unlimited number of such quotas may be specified.

The selection criteria compute a numerical index for each stand based on a sub-component or combination of sub-components. There are three variants of the criteria: one is based on stand age, the second computes the total yield of the selected sub-components, and the third computes the change in yield that would occur over the next 10-year period. Each operates in a similar manner, computing a numerical index for each stand that is subsequently used to sort the entire stand list. When multiple selection criteria are specified, each is applied sequentially and is used to select stands for harvest until a sub-quota, specified with the criteria, is met. The sub-quota specifies how much volume of the selected components must be harvested before the next criterion comes into effect. Each time a new selection criterion is applied a new index value is computed and the entire stand list is re-sorted. All yields from stands scheduled for harvest are tallied against the harvest quotas, regardless of the components specified in the selection criteria. Harvest scheduling stops when all of the harvest quotas have been satisfied, even if some of the selection criteria remain unused. Chapter 4 describes the scheduling process in greater detail.

After harvest scheduling is complete, the model simulates the allocation and scheduling of silvicultural treatments. Silvicultural treatment scheduling is limited by a user-specified maximum area of land that can be treated annually within a simulation period. The state-dependent rules describe regeneration alternatives and the new stand conditions that would result from the applying a treatment. In the present version of the HSG model, silvicultural treatments are limited to renewal treatments on stands harvested within the current simulation period. Similar to the harvest scheduling rules, the stands pending treatment are sorted based on a set of rules specified in the treatment priority list. Allocation of treatments begins at the top of the sorted list, and continues until the limit on total area has been exceeded. As each silvicultural activity is tallied, a record is written to the schedule file. After the silvicultural limit has been exceeded, the remaining stands are left untreated. These stands will regenerate in the next iteration using the natural regeneration rules of the state transition table.

Program output

The final step is to summarize the activities and changes that have occured in the current iteration. Summary information is written to a database that may be examined during or after program execution. A large volume of information is recorded describing forest composition, current changes, and the impact of the harvest treatments. Separate utility programs are provided to subdivide and format the summary data into a form suitable for analysis and presentation. The output files and the display programs are described in Chapter 5.

As mentioned previously, each simulation run produces a dataset that describes the schedule of management activities assembled by the model. Each record in the schedule database describes one harvest or silviculture activity, and contains information describing the unique stand identifier, the current year of the operation, the type of treatment, and the stand condition at the time of the treatment. The schedule may be printed or it may be processed by another program (for example, to compute average harvest cost). In addition, the schedule file can be linked to a GIS using the relational database management system, and the harvest schedule may be mapped out. One useful mapping theme is "Year of Harvest". Each operating period of the plan is assigned a colour code, and stands scheduled for harvest within each operating period are coloured accordingly (Figure 2.6). Visual examination of the harvest schedule map may reveal occasions where the proposed harvest schedule violates known operational constraints. The schedule may be amended (through the mandatory schedule or the operational constraint files), the simulation repeated, and the results reexamined.

At any time during the simulation, the model can be instructed to record the current condition of its modelling database into a system file for later examination. The file will contain a "snapshot" of the estimated state of each forest stand at that time. The "snapshot" would integrate changes to the forest that have occurred due to scheduled management treatments (such as harvesting or silviculture), as well as those due to growth and development. The "snapshot" file is saved in the same format as the original input stand list and contains the same kinds of information as the original file. Because the "snapshot" file is in the relational database format, it may be processed by other programs (say, to compute future wildlife habitat indices). Using the GIS, a time series of "snapshot" files (or their derivatives) could be mapped in order to provide a visual indication of the impact of the forest management activities (see Figure 2.7).

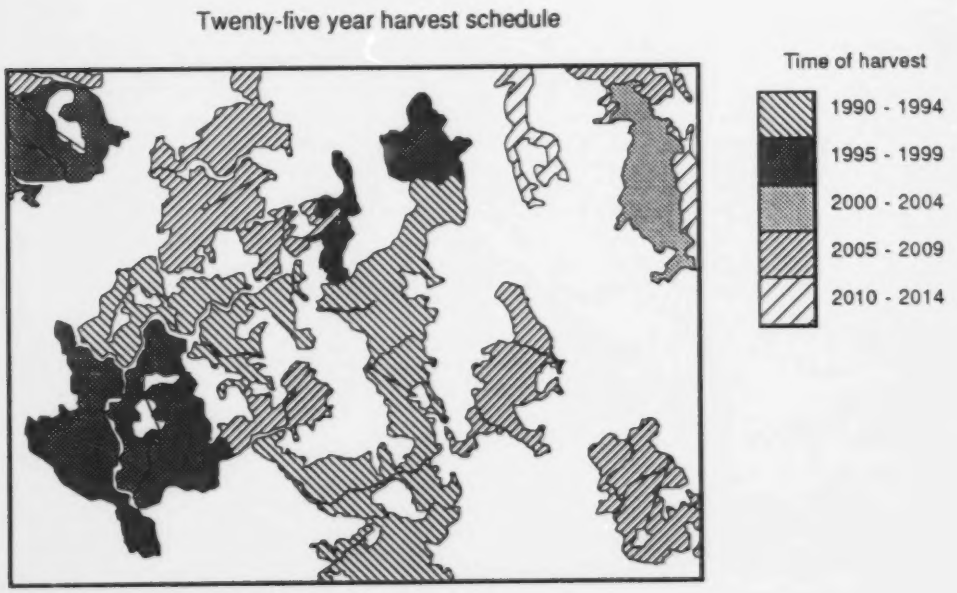


Figure 2.6. Portion of map showing the location of management activities scheduled by the wood supply model. The schedule was generated by the model and mapped using the GIS.

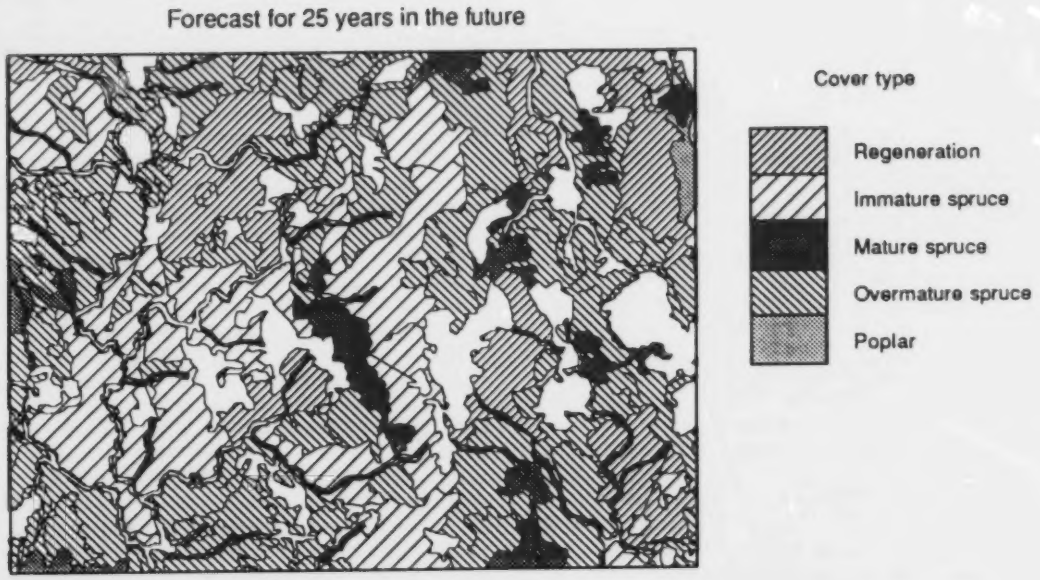


Figure 2.7. "Snapshot" maps show the state of the forest at any point in time during the simulation. This map forecasts the forest after implementing the first 25 years of the harvest schedule shown in Figure 2.5. The snapshot files are generated by the model and mapped using the GIS

Chapter 3 Dataset preparation

This chapter will describe the format and contents of the dataset required by the harvest schedule generating (HSG) model, and some of the procedures developed to assist in preparing the required datasets. A flowchart of input and output datafiles used in each step of processing is shown in Figure 3.1.

Stand list

The stand list contains one record for every productive forest stand in the study area. Other records may be included in the stand list (such as non-forest or non-productive forest), but these will not influence the simulation results and may increase computer processing requirements. Each stand record must be uniquely identified so that the GIS is able to match the stand records to the spatial description. The stand list record provides a 13 character field for unique stand identification. This has been used in the Iroquois Falls case study to form an identification number from a nine digit UTM map number plus a four digit stand number.

There are two parts to each stand list record. The first part provides identification and a brief description of the general characteristics of the stand. A pictorial representation of the stand list record is provided in Figure 2.4. An INFO record description is provided in Table 3.1. The meaning of each field is described in Table 3.2.

The second part of the stand list record contains a multiple number of detailed sub-component descriptions. Because the DBMS requires the stand list records to be of fixed length, the number of sub-components stored in the datafile must be fixed at some arbitrary limit. In the current version of HSG the stand list record has been set to contain five sub-component descriptions, resulting in a stand list record that is 94 bytes in size. To increase the number of initial sub-components, the HSG source file must be modified and recompiled.

Yield look-up tables

The yield look-up table file contains sets of curves (usually empirical yield curves) that describe the development of individual sub-component classes through time. A sub-component class is defined by a unique combination of the species code and the site-class code. The development pattern for each sub-component is modified by a "relative stocking value" (recorded in the stand list record for each sub-component). The relative stocking value will increase or decrease the yield value retrieved from the lookup table by a constant proportion, thus the shape of the develop-

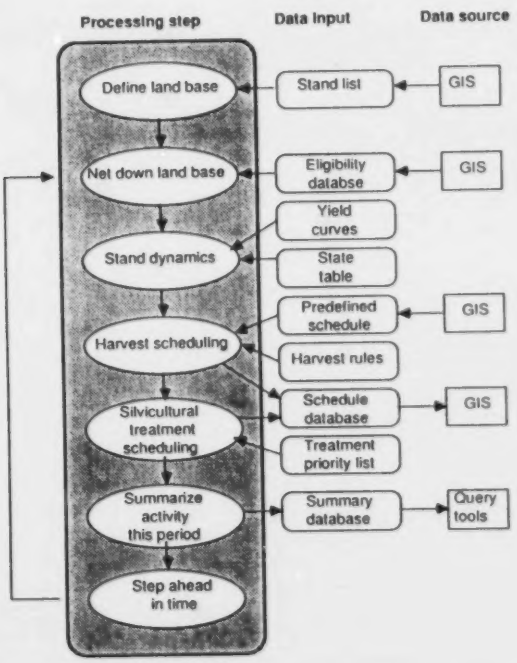


Figure 3.1. Flowchart of operations within the HSG model.

Table 3.1 INFO file definition for the stand table. This DBMS data dictionary describes the field names, field widths, and data types expected by the HSG model. See the INFO Reference Manual (Henco 1985) for further information about data types and formats.

COL	ITEM NAME	WDTH	OPUT	TYP	N. DEC	ALTERNATE NAME
1	STANDNUM	13	13	C	-	Stand-Number
14	ST-SUF	1	1	C	-	Suffix-Code
15	AREA	2	4	B	-	Area-in-Hectares
17	WG	2	2	C	-	Working-Group
19	SITE	2	2	C	-	Site-Class
21	YEAR	2	4	B	-	Year-of-origin
23	STATUS	2	2	B	-	Harvest-Eligibil
25	DISTURB	2	2	B	-	Disturbance
27	TREAT	2	2	B	-	Treatment-code
29	DATE	2	4	B	-	Date-of-Inventor
31	WGAGE	2	2	B	-	WorkingGroup-Age
33	SPARE	2	2	B	-	Spare-Field
35	CODE1	2	2	C	-	
37	SITE1	2	2	C	-	
39	YEAR1	2	4	B	-	
41	STKG1	2	4	B	-	
43	VOLUME1	4	3	F	0	
47	CODE2	2	2	C	-	
49	SITE2	2	2	C	-	
51	YEAR2	2	4	B	-	
53	STKG2	2	4	B	-	
55	VOLUME2	4	3	F	0	
59	CODE3	2	2	C	-	
61	SITE3	2	2	C	-	
63	YEAR3	2	4	B	-	
65	STKG3	2	4	B	-	
67	VOLUME3	4	3	F	0	
71	CODE4	2	2	C	-	
73	SITE4	2	2	C	-	
75	YEAR4	2	4	B	-	
77	STKG4	2	4	B	-	
79	VOLUME4	4	3	F	0	
83	CODE5	2	2	C	-	
85	SITE5	2	2	C	-	
87	YEAR5	2	4	B	-	
89	STKG5	2	4	B	-	
91	VOLUME5	4	3	F	0	

ment pattern will remain the same as the original curve although its height will vary (see Figure 2.3).

The yield curve sets are stored in column format in a system file (see Table 3.3). Each column describes the yield for one sub-component class, the leftmost column of the dataset lists the ages for each row, and column data are separated by tab characters. This format was selected because of the ease with which the data could be imported and exported to microcomputer spreadsheet packages. The yield curves can be analysed and assembled using a spreadsheet program, and then easily copied into a system file for use by the HSG program.

State table

The state table contains the rules which alter stand and sub-component descriptions. Each record in the state table file describes a single rule, and each of these rules is composed of two parts: a matching specification and an action specification. At the beginning of every simulation period, each record in the stand list is compared to the rules in the state table. When a match is found in the table, the associated action is applied to update the stand description.

The rules in the state table must be designed in the same manner as the yield curves. These rules embody

Table 3.2 Meaning of each field in the stand list file.

Field name	Meaning
STANDNUM	13 character stand identification number. Each stand record must have an unique identification number. The stand identifier could be composed from the mapsheet number and the stand sequence number.
ST-SUF	Suffix code. Not used by HSG model. May be used by DBMS to describe stand operational characteristics.
AREA	Stand area in hectares.
WG	2 character alphanumeric working group code. The working group code describes a silvicultural treatment class under which the stand will be managed.
SITE	2 character alphanumeric site class code. Used to describe the yield class or productivity of the site.
YEAR	Year of stand origin. When subtracted from the simulation year yields the age of the stand.
STATUS	Eligibility status of the stand. Not required as input to the HSG model. This field will be updated as the model operates.
DISTURBANCE	Management disturbance that occured in the current simulation period. Not used as input to the HSG model. This field will be updated as the model operates.
TREAT	Silvicultural treatment that was applied during the current simulation period. Not used as input to the HSG model. This field will be updated as the model operates.
WGAGE, SPARE	Spare fields used to hold derived information. These fields are used to contain derived data values generated by a DBMS program. These values subsequently mapped as themes using the GIS.
CODE1, CODE2, . .	2 character alphanumeric species codes describing component type.
SITE1, SITE2, . .	2 character alphanumeric site class code.
YEAR1, YEAR2, . .	Component year of origin.
STKG1, STKG2, . .	Component stocking level.
VOLUME1, VOLUME2 . . .	Calculated component yield. This value is generated by the HSG program from the yield look-up table for CODE and SITE, for a given age, multiplied by a stocking factor.

an empirical understanding of successional patterns of stand development, and thus must be developed to fit local forest conditions. As with the yield curves, the rules contain no biological understanding, but are merely a convenient and flexible means of applying empirical knowledge in the process of forecasting the growth and development of individual forest stands.

A sample of part of a state table is shown in Table 3.4. The file is in ascii format, and may be modified with a text processor. The first row in the state table file is a header record used for column identification, and is skipped over by the program. The header record provides a name label for each data column, and columns are separated by tab characters. The matching specification is separated from the action specification by a tab-separated "/" character.

Table 3.3 Sample of the format of the yield curve file. Each species code and site and class combination defines a sub-component pattern. The leftmost column defines the ages to which the data apply. This column formatted dataset can be easily imported from microcomputer spreadsheet packages.

Age	Sb X	Sb 1	Sb 2a	Sb 2b	Sb 3	Sw X	Sw 1
20	0	0	0	0	0	0	0
25	14	0	0	0	0	14	0
30	40	0	0	0	0	40	0
35	64	12	0	0	0	64	12
40	85	29	0	0	0	85	29
45	105	45	4	4	0	105	45
50	123	61	17	17	0	123	61
55	139	74	30	30	0	139	74
60	152	87	41	41	0	152	87
65	164	98	52	52	1	164	98
70	174	109	62	62	11	174	109
75	182	117	72	72	19	182	117
80	188	125	80	80	27	188	125
85	192	131	88	88	35	192	131
90	194	137	94	94	42	194	137
95	162	141	100	100	48	162	141
100	129	143	105	105	54	129	143
105	97	145	109	109	59	97	145
110	65	145	113	113	64	65	145
115	32	144	115	115	68	32	144
120	0	141	117	117	72	0	141
125	0	138	118	118	75	0	138
130	0	133	118	118	77	0	133
135	0	127	117	117	79	0	127
140	0	120	115	115	80	0	120
145	0	111	113	113	81	0	111
150	0	101	109	109	81	0	101
155	0	90	105	105	81	0	90
160	0	78	100	100	80	0	78
165	0	0	94	94	78	0	0
170	0	0	87	87	76	0	0
175	0	0	80	80	74	0	0
180	0	0	71	71	71	0	0
185	0	0	62	62	71	0	0
190	0	0	52	52	71	0	0
195	0	0	41	41	71	0	0
200	0	0	30	30	71	0	0

Matching specification

The matching specification contains six fields. A field may contain a specific data value or a "*" character. When a data value is used a match will be made only to a stand record having the same value. A "*" is a wild character, matching any value in the corresponding field of the stand record.

The first three fields match the working group, site-class, and age fields from the stand record. Note that the stand working group associates the stand with a set of silvicultural regimes, and is independent of the species codes used in sub-component descriptions.

The fourth field matches the relative stocking of the poplar sub-component. The matching criterion for this column specifies a relational operation that matches when the percent composition is less than, equal to, or greater than a specified value. Table 3.5 shows the format of the relational operators. A match occurs when the relative stocking field of the "Po" sub-component matches the relational operator. If the stand does not contain a "Po" sub-component description, the percent composition of "Po" is assumed to be 0. This is one of the few data-dependent aspects of the HSG model. If the criterion should match a species code other than "Po",

Table 3.4 Format and contents of a state table. This table shows the state table definition for one working group. Similar definitions would be entered for other working groups.

WG	Site	Age	%Po	distrb	Treatment		Sp1	St1	Age1	Stk1	Sp2	St2	Age2	Stk2
Sb	MX	*	*	CLEARCUT	INTENSIVE	/	Sb	MX	0	1.0				
Sb	X	*	*	CLEARCUT	INTENSIVE	/	Sb	MX	0	1.0				
Sb	X	*	=0	CLEARCUT	BASIC	/	Sb	X	0	1.0				
Sb	X	*	=0	CLEARCUT	EXTENSIVE	/	Sb	X	0	0.3				
Sb	X	*	>0&<31	CLEARCUT	BASIC	/	Sb	X	0	0.5	Po	1	0	0.5
Sb	X	*	>0&<31	CLEARCUT	EXTENSIVE	/	Po	1	0	1.0				
Sb	X	*	>30	CLEARCUT	BASIC	/	Po	1	0	0.65	Sb	X	0	0.4
Sb	X	*	>30	CLEARCUT	EXTENSIVE	/	Po	1	0	1.0				
Sb	X	>120	=0	NONE	NATURAL	/	Sb	X	20	0.8				
Sb	X	>120	>0&<31	NONE	NATURAL	/	Sb	X	10	0.3	Po	1	15	0.7
Sb	X	>120	>30	NONE	NATURAL	/	Po	1	15	0.8	Sb	X	10	0.2
Sb	1	*	*	CLEARCUT	INTENSIVE	/	Sb	M1	0	1.0				
Sb	1	*	=0	CLEARCUT	BASIC	/	Sb	1	0	1.0				
Sb	1	*	=0	CLEARCUT	EXTENSIVE	/	Sb	1	0	0.3				
Sb	1	*	>0&<31	CLEARCUT	BASIC	/	Sb	1	0	0.65	Po	2	0	0.4
Sb	1	*	>0&<31	CLEARCUT	EXTENSIVE	/	Po	2	0	1.0	Sb	1	0	0.2
Sb	1	*	>30	CLEARCUT	BASIC	/	Po	1	0	0.6	Sb	1	0	0.5
Sb	1	*	>30	CLEARCUT	EXTENSIVE	/	Po	1	0	1.0	Sb	1	0	0.1
Sb	1	>170	=0	NONE	NATURAL	/	Sb	1	20	1.0				
Sb	1	>170	>0&<31	NONE	NATURAL	/	Po	1	30	0.6	Sb	1	20	0.5
Sb	1	>170	>30	NONE	NATURAL	/	Po	1	40	0.8	Sb	1	20	0.3
Sb	2a	*	=0	CLEARCUT	EXTENSIVE	/	Sb	2a	0	0.7				
Sb	2a	*	=0	CLEARCUT	BASIC	/	Sb	2a	0	0.8				
Sb	2a	*	=0	CLEARCUT	INTENSIVE	/	Sb	MA	0	1.0				
Sb	2a	*	>0	CLEARCUT	EXTENSIVE	/	Po	3	20	0.7	Sb	2A	0	0.5
Sb	2a	*	>0	CLEARCUT	INTENSIVE	/	Sb	MA	0	1.0				
Sb	2a	>190	=0	NONE	NATURAL	/	Sb	2a	55	1.0				
Sb	2a	>190	>0	NONE	NATURAL	/	Sb	2a	50	0.7	Po	3	20	0.5
Sb	2b	*	*	CLEARCUT	EXTENSIVE	/	Sb	2b	0	0.8				
Sb	2b	*	*	CLEARCUT	BASIC	/	Sb	2b	0	0.9				
Sb	2b	*	*	CLEARCUT	INTENSIVE	/	Sb	2b	0	1.0				
Sb	2b	>220	*	NONE	NATURAL	/	Sb	2b	60	1.0				
Sb	3	*	*	CLEARCUT	BASIC	/	Sb	3	0	1.0				
Sb	3	>170	*	NONE	NATURAL	/	Sb	3	180	*				

the source code must be edited and the program recompiled.

Table 3.5 Format and meaning of the relational operators used in the state table. These relational operators may only be used in combination with the age and Pct fields.

Relational operator	Meaning
=0	match where value equals zero
0&	match where value is greater than zero and less than 31
30	match where value is greater than 30
*	match any value

The fifth field pertains to a disturbance code generated by the HSG program. Each stand record has a disturbance code which may be "CLEARCUT" if the stand was just harvested in the current iteration or "NONE" otherwise. The disturbance code is used to distinguish post-cut treatments from stand breakup or successional changes.

The sixth and final field in the matching specification is a treatment name, which identifies the type of management treatment that the rule applies to. The treatment field will only have a value if the distrubance code is not "NONE". If the disturbance code is "CLEARCUT", the treatment code and the rule apply to post-harvest treatments. After harvest, the silvicultural sheduling rules assign renewal treatments to each stand. The treatment name is restricted to one of the names specified in the treatment priority list file, or if the silvicultural quota has been exceeded, the treatment "NATURAL" is assigned. Treatment names used in the state table and treatment priority list may be any label (up to 20 character in length) that describes a regime, such as "INTENSIVE", "BASIC", or "NATURAL". When the stand update process encounters a stand that matches a disturbance code of "CLEARCUT", it searches the state table for a matching working group, site-class, and treatment code. When a match is found, the action part of the rule is applied.

Action specification

The action specification describes the changes that should be applied to the stand records and sub-component descriptions when a match occurs. The action specification is composed of a sequence of sub-component descriptions. A sub-component description is a set of four fields, each field separated by a tab character. Any number of sub-component changes may be specified, and the action specification ends when the end of the record is encountered.

The four fields in the action specification describe the species code, site-class code, age, and relative stocking value that should be set in the sub-component description. When a match is found between a state table rule and a stand record, the sub-components in the stand record are replaced by the sub-components described in the action specification. Any sub-components found in the stand record that are not also in the action specification are deleted. Any sub-components found in the action specification that were not previously in the stand record are added. This provides a mechanism to add, delete, or modify sub-component descriptions.

The species and site-class fields must contain exact data values. The age and stocking fields may contain exact values, or the "*" wildcard character. If an exact data value is used, it replaces the value previously held by the sub-component. If a wildcard character is used in the age or stocking field, the value previously held in that sub-component field is left unchanged.

Special characteristics of the state table

The first sub-component action specified in each rule is used to update the working group code, site-class code, and year of origin fields in the stand record. This ensures that stands will always have a sub-component with the same species code as the working group.

The order in which records are placed in the state table is important. The state table is searched from top to bottom until a match is found. When using wildcard codes it is possible to have multiple matching records in the state table. In cases where multiple records match, the HSG program uses the first match encountered and terminates its search. Generally, wildcard matches should be placed after exact matches of a similar type.

Frequently matched records should be placed at the top of the table in order to reduce the number of comparisons required. When loading the table, the program creates a tree-like data structure to hold the matching specifications. The most frequently occurring working group should be at the top of the table, followed by the next most frequently occurring. Within each working group, site classes should be sorted by the most frequently occurring to the least frequently occurring.

A program has been developed to format and print the state table as a tree-like structure. This format is useful for checking the state table specification for errors. The program is executed using the following UNIX command:

```
treemap < state_table > psfile
```

In the above command `state_table` should be replaced with the system file name of the state table file, and `psfile` should be replace with the system file name to be given to the output file. The `psfile` output file may be printed on a Postscript laser printer.

Figure 3.2 Representation of the state table as a decision tree. This figure was generated with the "`treemap`" program and printed on a Postscript printer.

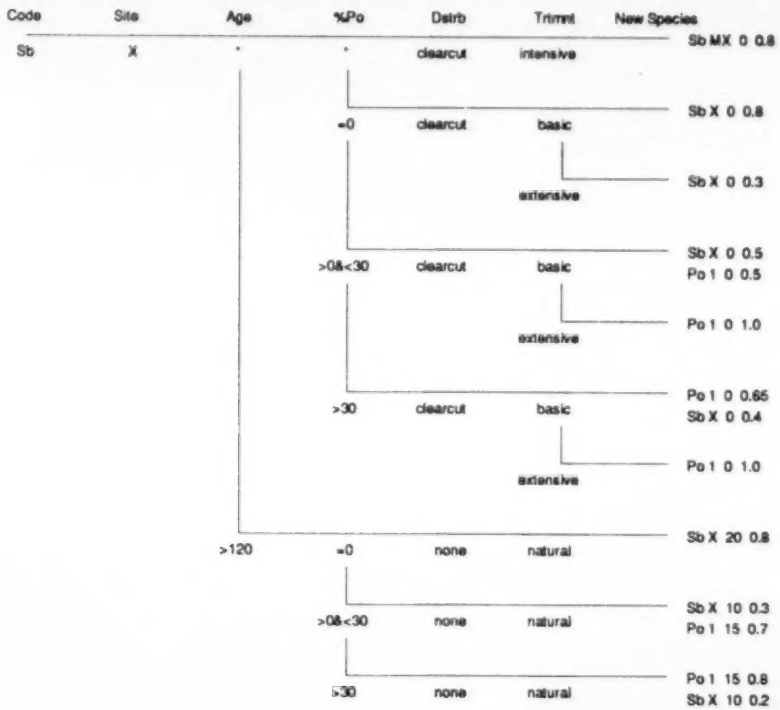


Figure 3.2 shows a sample of output created from the state table description shown in Table 3.4.

Treatment priority list

The treatment priority list contains a number of records, each specifying a treatment that should be applied to stands of a specific working group and site class combination. This list is used by the silvicultural treatment scheduling algorithm, which works in the following manner. Within the HSG program a limit to the annual silvicultural treatment area may be specified. Silvicultural treatments may be applied to recently harvested stands as long as the area of all treatments does not exceed the limit. The silviculture scheduling algorithm starts by reading the treatment priority list, and all stands that have been harvested in the current operating period are then sorted according to the order of working group and site-class codes encountered in the treatment priority list. Allocation of silvicultural treatments begins with stands at the top of the sorted list, and continues until the area treated exceeds the specified limit. The working group, site-class and treatment name fields are matched to a corresponding silvicultural treatment record in the state table. The contents of the matching state table record redefine the stand record and species sub-component fields. All stands harvested during that period that are left untreated regenerate according to the natural regeneration rules in the state table.

Records in the treatment priority list contain three fields, each separated by a blank or tab. The first field contains the working group code that the record should match, the second field contains the site-class code that the record should match, and the final field contains a treatment name that matches to a treatment specified in the state table. A sample of a treatment priority list is shown in Table 3.6.

Predefined schedule

The predefined schedule file is an input file that has a format similar to the stand list file (Table 3.7) and is identical to the output schedule file. Predefined activities define a specific sequence of harvest or silvicultural treatment activities that must be applied by the harvest scheduling algorithm. These activities are scheduled by the model regardless of the harvest quotas or selection criteria that are in effect. Using the predefined schedule file, arbitrary harvest sequences may be evaluated and compared to others generated arbitrarily or by the model.

The predefined schedule file must be sorted by year of treatment and stand identification number. The

Sb	MX	INTENSIVE
Sb	X	INTENSIVE
Sb	1	INTENSIVE
Sb	M1	INTENSIVE
Sb	2	INTENSIVE
Sb	2a	INTENSIVE
Sb	MA	INTENSIVE
Sb	2b	BASIC
Sb	3	BASIC
Sw	MX	INTENSIVE
Sw	X	INTENSIVE
Sw	1	INTENSIVE
Sw	2	INTENSIVE
Sw	2a	INTENSIVE
Sw	2b	BASIC
Sw	3	BASIC

Table 3.6 Format and sample contents of the treatment priority list. This list contains one record for each working group and site class combination to be treated. Each record contains the name of a treatment found in the state table.

predefined schedule file is checked during each simulation step after sub-component volumes have been updated, and before harvest scheduling criteria are applied. The model reads records from the predefined file that have dates earlier than or equal to the current simulation date. As each record is read in the working group, site class, year of origin, and sub-component fields are compared to matching fields in the stand record with the same stand identification number. If the specified fields do not match, the predefined treatment is considered invalid, and is not applied. If the fields match, the harvest or silviculture treatment specified in the predefined record is carried out. As the treatment is carried out, the volumes harvested or areas treated are tallied against any quotas or limits that might apply. Stand eligibility status is not checked when applying predefined activities.

As mentioned previously, the predefined schedule file has a format similar to the stand list file and is identical to the output schedule file. There are two easy methods of generating a predefined schedule file. The first method uses the DBMS and the stand list file. Using the DBMS, records describing stands to be harvested are reselected in the stand list file and copied to the predefined schedule file. The treatment code field must be filled in to describe the treatment to apply to each stand. The date field must also be updated to define the time in the future when the treatment should be applied. A predefined treatment schedule created in this manner must be sorted by date and stand identification number before being input to the model.

Another easy means of generating a predefined schedule file involves use of the DBMS and an output schedule file. In this method the HSG model is run first to generate a schedule file, based on appropriate mill demands and harvesting rules. The DBMS is used to edit the schedule file, changing the dates of some of the harvest or silviculture treatment activities and possibly adding or deleting some records. After editing, the list must be sorted by year of treatment and stand identification number.

A predefined treatment schedule may be used alone or in combination with the automatic harvest and silviculture scheduling rules. When used alone only the treatments specified in the predefined file are applied. When used in combination, the automatic rules are applied second and "top up" the harvest schedule to satisfy the quotas in effect. If the quotas are satisfied by the predefined schedule, no additional stands are scheduled.

Eligibility database

The eligibility database provides a means to describe when stands become ineligible, or re-eligible, for harvest scheduling. Initially all stands are considered eligible for treatment scheduling. The eligibility database is read at each simulation step during the stand update process. As records are encountered in the eligibility database they are matched to corresponding stand records. The eligibility records specify dates when stands should be marked as ineligible, or re-eligible, for treatment scheduling. A flag in the stand record is used to contain the current status. If a stand is marked as ineligible, it is bypassed by the harvest scheduling algorithm.

The eligibility database consists of one record for each change to the status of a stand's eligibility. Each record has two fields: the first is the stand identification number, and the second is the year when the change should be applied. Dates prefixed with a minus set the status to ineligible. Dates without the minus prefix set the status to eligible. The eligibility list should be sorted by the absolute value of year and by stand identification number.

Although simple in format, the eligibility database is not convenient to work with. A more convenient format has been defined, along with a program to convert that format to the eligibility database format. The more convenient format consist of a file with three fields. Each record defines a period when a stand is ineligible for harvest scheduling. The fields on the record define the stand identification number, the starting year of the ineligible period, and the ending year of the ineligible period. In this way a single record states the time to flag the stand ineligible and the time to flag the stand re-eligible (see Table 3.8).

A GIS can be used to prepare a file describing management constraints. For example, a buffer command can be used to identify stands within a set distance of a feature of interest and the attribute records of

Table 3.7 **INFO definition for the pre-defined schedule file. This file definition is similar to the stand table file and identical to the output schedule file. The only difference from the stand list file is the interpretation of the DATE field. In this file DATE contains the year which the treatment should be applied.**

COL	ITEM NAME	WDTH	OPUT	TYP	N. DEC	ALTERNATE NAME
1	STANDNUM	13	13	C	-	Stand-Number
14	ST-SUF	1	1	C	-	Suffix-Code
15	AREA	2	4	B	-	Area-in-Hectares
17	WG	2	2	C	-	Working-Group
19	SITE	2	2	C	-	Site-Class
21	YEAR	2	4	B	-	Year-of-origin
23	STATUS	2	2	B	-	Harvest-Eligibil
25	DISTURB	2	2	B	-	Disturbance
27	TREAT	2	2	B	-	Treatment-code
29	DATE	2	4	B	-	Date-of-Treatmen
31	WGAGE	2	2	B	-	WorkingGroup-Age
33	SPARE	2	2	B	-	Spare-Field
35	CODE1	2	2	C	-	
37	SITE1	2	2	C	-	
39	YEAR1	2	4	B	-	
41	STKG1	2	4	B	-	
43	VOLUME1	4	3	F	0	
47	CODE2	2	2	C	-	
49	SITE2	2	2	C	-	
51	YEAR2	2	4	B	-	
53	STKG2	2	4	B	-	
55	VOLUME2	4	3	F	0	
59	CODE3	2	2	C	-	
61	SITE3	2	2	C	-	
63	YEAR3	2	4	B	-	
65	STKG3	2	4	B	-	
67	VOLUME3	4	3	F	0	
71	CODE4	2	2	C	-	
73	SITE4	2	2	C	-	
75	YEAR4	2	4	B	-	
77	STKG4	2	4	B	-	
79	VOLUME4	4	3	F	0	
83	CODE5	2	2	C	-	
85	SITE5	2	2	C	-	
87	YEAR5	2	4	B	-	
89	STKG5	2	4	B	-	
91	VOLUME5	4	3	F	0	

Table 3.8 INFO defintition for the management constraint file. Many GIS layers may be used to create a number of constraint files. The files are then appended and processed using the `constprep` command.

COL	ITEM NAME	WDTH	OPUT	TYP
1	STANDNUM	14	14	C
15	START-YEAR	4	4	I
19	END-YEAR	4	4	I

these stands can be marked with a time period when harvest is prohibited; multiple GIS layers may be created in this way (see Figure 3.3). The attribute records for the multiple layers can be appended together, and then processed to convert the records to the eligibility database format. The processing program adjusts the records to account for overlapping ineligibility periods.

The following UNIX command shows the format of the eligibility database preprocessing command:

```
constprep <constraints >eligible
```

In the above example, `constraints` should be replaced with the name of a system file created by appending attribute files containing lists of ineligible periods. `Eligible` should be replaced with the name of the system file that will contain the resulting eligibility database.

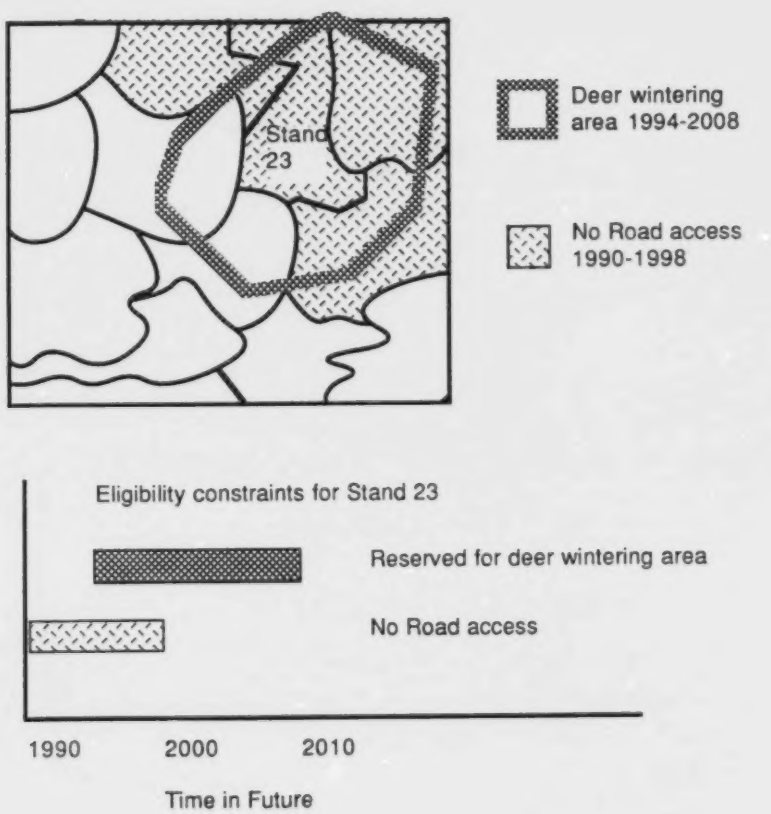


Figure 3.3. Multiple GIS layers may be used to create the eligibility database. Overlapping constraint periods are merged by the `constprep` program.

Chapter 4 The Harvest Schedule Generator

This chapter describes procedures to invoke and control the execution of the Harvest Schedule Generator (HSG).

Invoking the HSG model

Execution of the HSG program is controlled by a set of statements contained in a parameter file. The parameter file is read through the standard input stream and, as the program executes, informative messages and errors are printed on the standard error stream. Summary data describing the state of the forest through the course of the simulation are written on the standard output stream. The following example shows a UNIX command that would invoke the HSG model:

```
hsg <parameter >summary
```

In the above example `parameter` is the name of the input parameter file and `summary` is the name of the output summary file. Other files are used as input and output by the model. The names of these files are specified using statements in the parameter file.

Contents of the parameter file

Each line of the parameter file contains a single HSG command. The general syntax of all commands is that the command keyword is left justified and typed in upper case. Command parameters are separated by spaces and are placed on the right of the command keyword. There is no limit to the number of command lines that may appear in a parameter file.

Numeric operands are specified as either integer or fixed decimal numbers. However, in some cases fixed decimal numbers would not make sense (for example, a fractional time interval). Integer numbers should be used unless this manual explicitly states that fixed decimal numbers are permissible.

Care should be taken in specifying file names because the UNIX is case-sensitive with respect to file names. Unlike other operating systems, the UNIX file names `a.dat` and `A.DAT` refer to two separate files.

Within the parameter file, required and optional statements specify input datafile names, output datafile names, and modify the flow of program control. Table 4.1 lists the control statements used in the HSG program. The following sections will describe each command in greater detail.

Documentation statements

```
TITLE          descriptive line of text
```

The `TITLE` command provides a single line of descriptive information that identifies the simulation run. All

Table 4.1 Commands known to the HSG program. These commands are listed in the order that they would be specified in a parameter file.

Command	Type
TITLE	Optional documenting title
DESC	Optional documenting comment
INDEX	Optional input file
INVENTORY	Required input file
YIELD	Required input file
STATES	Required input file
TREATMENTS	Optional input file
CONSTRAINTS	Optional input file
MANDATORY	Optional input file
SCHEDULE	Optional input file
BEGIN	Required initialization statement
SILVA	Optional parameter
OPMIN	Optional parameter
STEP	Required control statement
SNAPSHOT	Optional output file

text following the title keyword is considered to be the title. Although not used by the HSG program, the title record is copied to the summary file and may be used to label output from the query and display programs.

```
DESC          descriptive text
```

The description records provide a means to document the purpose of a simulation run. Any number of DESC records may be entered in a parameter file. The DESC records are not interpreted by the HSG program, but are copied to the summary file as a record of the simulation.

Required Input Files

```
INVENTORY     filename
YIELD         filename
STATES        filename
```

The above three statements are used to specify to the model the names of the required input files. INVENTORY specifies the name of the stand list data file. YIELD specifies the name of the data file containing the yield curve set. STATES specifies the name of the data file containing the state table. In the above examples filename would be replaced with an appropriate system file name. The format and contents of these data files are described in Chapter 3. These statements must be specified and must be placed before the BEGIN statement; if not, a fatal program error will occur. If a file name is specified but the file cannot be opened, a fatal program error will occur. No other data should occur on the line after the filename.

The INVENTORY statement may only be specified once, at the beginning of the run.

The YIELD statement may be specified several times. This provides a mechanism to change the defintion of the yield curves over time (for example, to simulate the impact of climate change on forest yield), or to split the yield look-up tables into several files. Each time the YIELD statement is specified, the HSG program will load the specified file. Data for previously defined species and site class combinations will replace the existing look-up data. Data for undefined species and site class combinations will be added to the look-up database.

The STATE statement may be specified several times. Each time the STATE statement is encountered the program will reload the state table from the specified file.

Optional Input files

```
INDEX         filename
```

The INDEX file is an optional file which associates numeric codes to the labels used in the state table. The state table refers to treatments by name, such as "INTENSIVE" or "EXTENSIVE", but these long character labels cannot be stored in the 2-byte field reserved for this purpose in the SNAPSHOT or SCHEDULE files. The INDEX file provides a means to convert the character labels to numeric codes, which can then be stored in the data files.

The INDEX file is an ascii file which may be edited using a text processor. Each line in the file specifies a name-number triplet. There are three columns in the file, and columns are separated by tab characters. The first column contains the name of the field to which the conversion applies, and must have a value of "disturb" or "treat". A value of "disturb" indicates that this record refers to labels in the disturbance field of the state table, and a value of "treat" indicates that this record refers to labels in the treatment field of the state table. The second column contains the character label, and the third column contains the numeric code. A sample of the index file is shown in Figure 4.2.

If the INDEX file is not specified, the HSG program will automatically convert the character labels to numeric codes. However, this conversion will be arbitrary and the meaning of the codes will be undefined.

```
CONSTRAINT    filename
TREATMENTS    filename
MANDATORY     filename
```

The above three statements specify to the model the names of the optional input files. CONSTRAINTS specifies the name of the eligibility database. TREATMENTS specifies the name of the treatment priority list. MANDATORY specifies the name of the predefined treatment schedule. The format and contents of these data files are described in Chapter 3. In the above examples, filename must be replaced with an appropriate system file name and no default value should be used. Specification of these commands are optional. The CONSTRAINTS and MANDATORY statements can occur only once and must be specified before the BEGIN statement. The TREATMENTS statement can be specified any number of times, and can occur anywhere in the parameter file. Each time the TREATMENTS statement is specified a new treatment priority list is loaded.

Note: If an optional input file name is specified, but cannot be opened, a fatal program error will occur.

Optional output files

```
SCHEDULE      filename
```

The SCHEDULE statement controls the recording of the harvest treatment schedule and if this line is omitted HSG will not produce a schedule file. The filename parameter is optional. If present, the HSG program will start recording the schedule using filename as the system file name. If a file name has not been specified, the HSG program will stop recording the treatment schedule. The format and contents of the schedule file

are described in Chapter 5. The SCHEDULE may be specified any number of times within an HSG parameter file. Each time the command is specified it will redirect the schedule output.

Note: The SCHEDULE statement causes the recording of treatments to be turned on and off. The schedule file can contain a large number of records. Disk space may be conserved by turning recording off for time periods when a record of the schedule is not required.

SNAPSHOT filename

The SNAPSHOT statement causes the model to dump out a current copy of the modelling database. The format and contents of the SNAPSHOT file are described in Chapter 5. The SNAPSHOT statement is optional, and may occur more than once. A SNAPSHOT statement may not be placed before the BEGIN statement. The placement of the SNAPSHOT command relative to the STEP statements dictates the time during the simulation that the snapshot file is created. Multiple SNAPSHOT commands may be specified, causing multiple dumps of the modelling database to be recorded. The filename parameter is optional. If specified, it provides the system file name to be used to store the data. If not specified, a file name of the form 'snapYYYY' will be used, where 'YYYY' is the last year of the current simulation period. If a snapshot is taken before the first simulation step, a copy of the inventory updated to the beginning simulation year is dumped.

Note: SNAPSHOT files are as large as the input inventory files and frequent use of the SNAPSHOT command may consume large amounts of disk space.

Initialization command

BEGIN date

The BEGIN serves two purposes. First, it specifies a date to be used as the first year of the simulation. All stand records are updated to this date before the simulation starts. All subsequent simulation steps occur relative to the initial starting date. Date must be specified as an integer parameter. The value of date is typically the current year when the simulation is started.

The second purpose of the BEGIN statement is to direct the HSG program to initialize the modelling database. The required INVENTORY and optional MANDATORY statements must be specified before the BEGIN statement is used. If the specified files cannot be opened the program will generate a fatal error.

Run time commands

The following commands affect the run time performance of the HSG model.

SILVA value

The SILVA statement is optional. If specified, value specifies the maximum annual area of silvicultural treatments that may be applied. Silvicultural treatments are specified in the treatment priority list and also in the state table. Value specifies a limit in hectares. If an unlimited amount of silviculture is permissible, specify an arbitrarily large number. If the SILVA statement is not specified or if the limit is exceeded in any simulation period, harvested stands will only regenerate according to the "NATURAL" regeneration rules specified in the state table.

Note: If "NATURAL" regeneration rules are not defined in the state table, stands will not regenerate after harvest when the SILVA limit has been exceeded.

OPMIN value

The OPMIN statement specifies a minimum operable volume that is computed by the yield calculation procedures. The OPMIN statement is optional. If not specified, no minimum operable volume limit will be used. If specified, the sum of all sub-component yields must be greater than the OPMIN value, or else all sub-component yields will be recorded as zero. Value is a fixed decimal number, and is specified in units of m^3 per hectare.

Note: All volumes in the database are recorded as operable volumes. Change to the OPMIN value may cause a change to volumes recorded in the summary, schedule, and snapshot files.

STEP step_size : harvest_quotas : rules

The step statement is the most important command in directing the course of the simulation. Each step statement directs the HSG model to simulate harvest, silviculture, and growth through one simulation step. The step statement specifies the size of the simulation steps, the amount of volume that the model should attempt to harvest each year, and the rules to be used by the harvest scheduling algorithm. As can be seen in the above example, these three components are specified on the same line as the step command, and are separated by colons. The following sections will describe each component in greater detail.

The step size is an integer number which specifies the number of years to be included in the current simulation period. Step size is added to the final year of the previous simulation period to determine the final year of the current simulation period. The modelling database is updated by forecasting stand growth and development up to the final year of the current period. All annual volume targets and limits are multiplied by the step size to determine periodic values and the final year of the current simulation period is recorded as the date of operation for treatments and activities scheduled within the simulation period.

Note: Small step sizes result in finer resolution in the schedule file and may possibly improve the growth forecasts. However, this must be balanced against the requirement for more steps, and therefore more computer resources, to cover the same simulation horizon as done with large steps.

The second group of parameters of the STEP statement are the harvest quotas. The quotas specify the amount of wood that the model must attempt to schedule for annual harvest in the current simulation period. Multiple quotas may be specified, and each quota may specify multiple species. The general syntax for specifying the harvest quota is as follows:

```
species=volume, species/species=volume

Example: Sb/Sw=85000, Po = 30000
```

In the above example, the species value is an alphanumeric code which corresponds to a sub-component code in the modelling database. `Volume` is a fixed decimal number, and is specified in units of annual yield of the sub-component (usually m^3 per year). If multiple quotas are specified, they must be separated by commas. If multiple species codes are specified they must be separated by "/" characters. When multiple species codes are specified, the quota will be satisfied by the total harvest from those species, regardless of the proportion; an example is given at the end of this chapter. There is no limit to the number of species codes or quotas that may be specified.

The third group of parameters on the step statement contains the rules used by the scheduling algorithm to prioritize stands for harvest. When multiple rules are specified each is applied sequentially until all of the harvest quotas have been satisfied for that simulation period. Specified with each rule is a sub quota and each rule remains in effect until its specified sub-quota has been exhausted. The general syntax of the harvest rules is as follows:

```
case-species=subquota(minvol),case-
species/species=subquota

Example: 2-Sb/Sw=85000(80), 2-Po=30000
```

As can be seen above, when multiple rules are specified each is separated by a comma and when multiple species codes are specified each is separated by a "/" character. There are several variations of the scheduling algorithm. The first part of the rule describes which case to use. Presently, three cases have been coded into the HSG model. Each is described in the following paragraphs.

Case 0 schedules stands for harvest based on an "oldest-first" ordering. Only stands belonging to working groups specified in the species code list are harvested and the stand list is sorted in decreasing order of age. Stands are harvested from the top of the list until the harvest quotas are satisfied, until the sub-quota is met, or until the stand list is exhausted. For case 0, the sub-quota specifies the maximum area (in hectares) to be harvested from the specified working groups using this rule.

Case 1 schedules stands for harvest based on a "highest current volume" rule. For each stand, the current operable volume for sub-components in the species code list is calculated. The stand list is then sorted in decreasing order based on the calculated volume. Stands are scheduled for harvest from the top of the list until the harvest quotas are satisfied, until the sub-quota is met, or until the stand list is exhausted. For case 1, the sub-quota specifies the maximum volume to be harvested from the specified sub-components using this rule.

Case 2 schedules stands for harvest based on a "minimize volume loss" rule. A numerical index is computed from the change in volume that is forecast to occur over the next 10-year period. Volumes are measured only for sub-components in the harvest rule species list. The stand list is sorted in ascending order based on the volume change. Stands are scheduled for harvest from the top of the list until the harvest quotas are met, the sub-quota is met, or the stand list is exhausted. For case 2, the sub-quota specifies the maximum volume to be harvested from the specified sub-components using this rule.

The `minvol` parameter is optional and specifies the minimum average yield that must be present in a stand in order for the stand to be considered for harvest. `Minvol` is specified as a fixed decimal number and represents a minimum operable volume limit of the prefered species. Any stand which has less than `minvol` yield of the prefered species is not eligible for harvest. If `minvol` is not specified, no minimum operable yield limit applies.

Annotated example of a parameter file

To clarify the specification of the HSG statements, an example is provided in Figure 4.1. This following section will describe the function of some of the statements shown in the example.

The first six lines in the file document the purpose of the simulation run. The contents of these lines will be written out to the summary file as a form of self-documentation.

Lines 7 to 12 in the sample file provide required and optional file names to the model.

Line 13 is the BEGIN statement, and specifies that the initial year of the simulation will be 1989. As the inventory file is loaded, all stand volumes will be updated to 1989.

```
 1  TITLE Run number 1. Test maximum sustainable yield
 2  DESC
 3  DESC This simulation run is one of a series of runs
 4  DESC used to estimate maximum sustainable yield on the
 5  DESC test forest dataset.
 6  DESC
 7  INVENTORY      /usr/forest/stand.dat
 8  YIELD          /usr/forest/yield.dat
 9  STATES         /usr/forest/state_table
10  TREATMENTS     /usr/forest/treatments
11  CONSTRAINTS    /usr/forest/access_constraints
12  SCHEDULE       /usr/scenarios/run1/schedule
13  BEGIN          1989
14  OPMIN 40
15  SILVA 1000000
16  STEP 5 : Sb/Sw=85000 : 2-Sb/Sw=10000(80),2-Sb/Sw=85000
17  STEP 5 : Sb/Sw=85000 : 2-Sb/Sw=10000(80),2-Sb/Sw=85000
18  STEP 5 : Sb/Sw=85000 : 2-Sb/Sw=10000(80),2-Sb/Sw=85000
19  STEP 5 : Sb/Sw=85000,Po/Pb=20000:2-Sb/Sw=85000,2-Po/Pb=20000
20  STEP 5 : Sb/Sw=85000,Po/Pb=20000:2-Sb/Sw=85000,2-Po/Pb=20000
21  SNAPSHOT /usr/scenarios/run1/snap2013
22  SCHEDULE
23  STEP 5 : Sb/Sw=85000,Po/Pb=20000:2-Sb/Sw=85000,2-Po/Pb=20000
24  STEP 10 : Sb/Sw=85000,Po/Pb=20000:2-Sb/Sw=85000,2-Po/Pb=20000
25  STEP 10 : Sb/Sw=85000,Po/Pb=20000:2-Sb/Sw=85000,2-Po/Pb=20000
26  STEP 10 : Sb/Sw=85000,Po/Pb=20000:2-Sb/Sw=85000,2-Po/Pb=20000
27  STEP 10 : Sb/Sw=85000,Po/Pb=20000:2-Sb/Sw=85000,2-Po/Pb=20000
28  STEP 10 : Sb/Sw=85000,Po/Pb=20000:2-Sb/Sw=85000,2-Po/Pb=20000
```

Figure 4.1. A sample parameter file. Some of the commands in this sample file are described in the text.

The OPMIN statement on line 14 specifies a minimum operable volume limit of 40 m^3 per hectare. Any stands with total yields of less than 40 m^3 per hectare will be set to have yields of 0 m^3 per hectare in all subcomponents. This value will effect the calculation of total operable growing stock, and may also influence the harvest schedule algorithm which bypasses stands with no volume.

The SILVA statement on line 15 specifies an annual limit of 1 000 000 hectares of silviculture. This large value allows the model to apply silvicultural treatments wherever possible. The OPMIN and SILVA statements only need to be specified once. They remain in effect for the rest of the simulation run or until they are changed.

Line 16 contains the first STEP statement. This statement indicates that the first simulation step should be a 5-year period. This statement also indicates that the model should attempt to harvest an annual volume of 85 000 m^3 of a combination of the Sb and Sw stand subcomponents. Over a 5-year period, this would amount to a 425 000 m^3 harvest. Two harvest rules are specified; both require the model to prioritize for harvest the stands that are declining the fastest (or growing the slowest) in combined Sb and Sw operable volume. The first rule attempts to harvest 10 000 m^3 per year from stands containing at least 100 m^3 per hectare of prefered species. After 50 000 m^3 are harvested (10 000 m^3 per year covering the 5-year period), or after the stand list is exhausted, the second rule takes over. This rule relaxes the minumum operable volume restriction and attempts to schedule enough stands for harvest to complete the harvest quota.

Lines 17 and 18 repeat the same management strategy for two more 5-year simulation periods.

The fourth step statement (line 19) directs the action for the fourth simulation period. This simulation period

also has a 5-year time step, and would cover the 15th to 20th year into the future. In this time step an additional harvest quota and rule was added for poplar. The second harvest quota is to harvest at least 20 000 m^3 per year of Po or Pb. The second rule indicates that, if the poplar harvest was not satisfied after the first rule, the stand list should be sorted by declining poplar volume, and harvested until the quotas are satisfied. This rule is repeated during the fifth time step (line 20).

After the fifth time step, a snapshot of the modelling database is copied to a system file (line 21). Line 22 indicates that the HSG model should stop recording the treatment schedule.

Line 23 is the sixth 5-year time step. At the end of this step 30 years of simulated time have past. The next five time steps statements each cover 10-year simulation periods. At the end of the last step statement (line 28) the simulation clock would have advanced a total of 80 years.

The summary file created by this run will contain the output from 11 time steps. The first six time steps cover 5-year periods, the next five time steps cover 10-year periods. The schedule file created by this run contains the harvest and silvicultural schedule, on a stand by stand basis, for the first 25 years of the simulation. No record will be kept of the schedule beyond the first 25-year period.

Chapter 5. Analyzing and Displaying Output

This chapter describes the output datasets generated by the model, and also describes procedures to interpret and display the output files. The model does not produce a formatted report of simulation results, instead, three data files (a summary data file, a schedule data file, and a snapshot file) are produced. These data files must be formatted and manipulated with other programs and several programs are provided to assist in this task.

Summary data

The HSG model produces summary output 'both aggregate and detailed' during all runs. In producing the summary data, all stands are aggregated together and spatial identity is no longer maintained. However, the data are summarized into a large number of categories of forest type and status, thus providing considerable detail on the overall forest condition.

The four main categories of data in the summary file are increment, mortality, operable growing stock, and stand area (see Figure 5.1). Each of these categories are further subdivided into categories of non-reserved, reserved, and harvested. The harvested sub-category refers to the stand area that was harvested in the current simulation step, or the operable volumes yielded by the harvest. The reserved sub-category refers to the forest area marked as ineligible for treatment during the current period, or the volumes of operable growing stock present on that area. The non-reserved sub-category refers to the areas of forest stands that were not harvested or marked as ineligible during the current period, and to the operable growing stock present in those stands.

Each of the above sub-categories is further broken down into sub-categories of working group, site class, and age class. This category hierarchy is defined in the program source code for the HSG model. Changes to this hierarchy can be made by reprogramming and recompiling the HSG program. However, class divisions within each sub-category are defined by the working group codes, species codes, and site-class codes that exist in the inventory dataset. New species and site class divisions are formed as the codes are encountered in the dataset. The model is independent of the input data in this respect, and there is no need to modify the source code to accommodate new species or site breakdowns.

The top level category in the summary file is the date of the operating period to which the summary data apply. A set of summary information is produced during each step of the simulation. Thus, a summary file from a 10-period simulation would contain 10 sets of summary data.

The summary data are expressed in units appropriate to the data that they represent. All increment and mortality categories are recorded in units of average m^3 per year for the period represented. Reserved and non-reserved categories of growing stock are recorded in units of total m^3 present at the end of the period after the effects of harvest have been simulated. Harvested growing stock is recorded in units of average

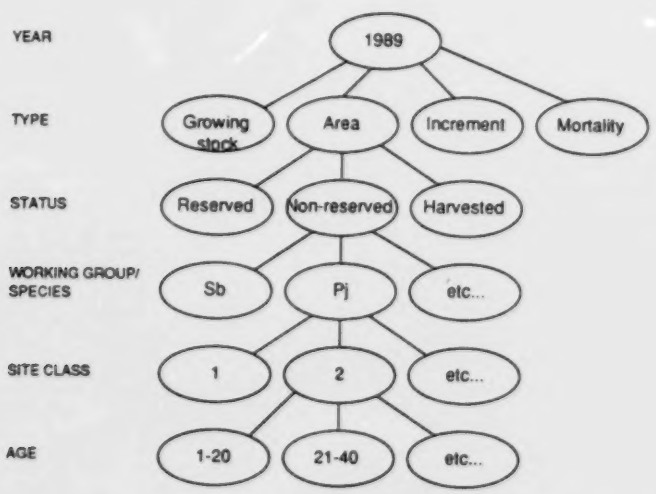


Figure 5.1. Categories of summary data. The summary data are subdivided according to the categories shown above. Only one branch of the tree has been shown for clarity, but all types and statuses are subdivided into species, site, and age class.

m^3 per year for the period represented. All area categories are represented in hectares.

The internal format of the summary file has been designed for portability, efficiency, and flexibility of data extraction. There is one record in the file for each of the lowest-level sub-divisions. This results in a file that has a large number of records for each periodic summary. Each record is composed of two parts: the first part contains the category data, the second part contains the numeric value for the specified category.

Although the summary file is composed entirely of readable ASCII characters, visual interpretation of the raw data would be difficult. However, several utility programs have been developed to allow versatile formatting and summary of the dataset. These programs have been designed to operate in a complementary manner, so that one program can be plugged into the next (see Figure 5.2). The programs each expect the same format of input file, and can be connected to each other in a variety of ways. The following sections describe the function of each program.

SUBSET

Syntax: `subset -i 'querystring' <summary >subfile`

The subset program is used to extract a limited set of category data from a summary file. This program acts as a filter by reading a summary file from standard input and writing a file in the same format to standard output. The command line to the program must also include a "query" that specifies the criteria to be used for the reselection. In the above UNIX example, `summary` would be the name of the input file, and `subfile` would be the name of the output file. The `querystring` would be replaced with an appropriately formatted query.

The following rules apply to the format of the `querystring`:

1) Compound clauses are seperated by colons.

2) Each clause is composed of a single category type and, optionally, one or more data codes.

3) If one or more data codes are specified, a relational operator must seperate the two parts. The following relational operators are allowed: "=" equal to; "<" less than; ">" greater than; "^" not equal to.

4) If two or more data codes are specified in a clause, they must be separated by an associative operator. The following two operators are allowed: "/" or; "+" and. Codes which are "or"ed together form seperate divisions in the output file. Codes which are "and"ed together form a single division in the output file.

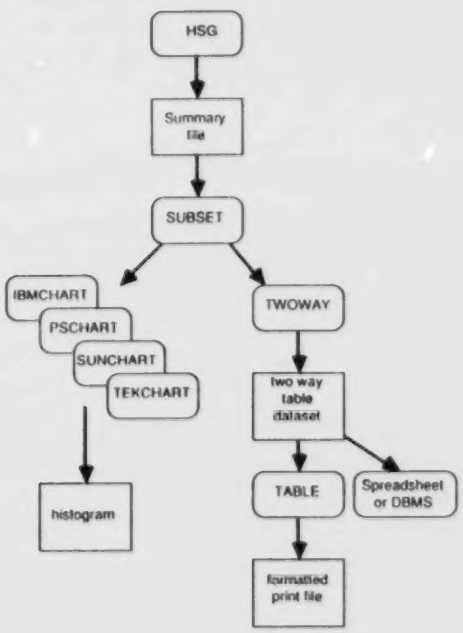


Figure 5.2. Programs for selecting and displaying information produced by the HSG model. The output from one program is used as input to the next.

5) If a category is specified but no data codes are supplied, all codes will be included in the output file.

6) The categories are case sensitive, and must be typed exactly as defined in the HSG program.

7) The data codes are case sensitive, and must be typed exactly as defined in the input data files.

8) Clauses are evaluated from left to right. Data in the output file are rearranged in the same order as the clauses.

9) Categories not present in the `querystring` will not be present in the output file.

10) The string should be surrounded by apostrophes.

Example:
```
'TYPE=volume:SPECIES:SITE=X+1/2+3'
```

The above query has three clauses. The first specifies that data will be summarized only for the major category of "`volume`". The second specifies the category of "`SPECIES`", but no codes. As a result, all codes for category "`SPECIES`" will be included in the output file. The third clause specifies that two divisions of site-class are required: one composed of site-class codes `X` and `1`, and a second composed of site-class codes `2` and `3`. No other site-class codes will be included.

As mentioned before, the categories are programmed into the HSG model. Only the categories listed in Table 5.1 can be used in a query. However, the codes are those found in the input datasets. No reprogramming is required to use a set of codes that apply to local inventory conditions.

Table 5.1. Categories defined in the HSG program and available in the summary file. Codes marked with "*" are defined by the datafiles being used and not coded into the programs. There is a "YEAR" code for every operating period that was simulated.

Category name	Available codes	Meaning
YEAR	*	operating period
TYPE	volume incr mort area	current growing stock periodic increment periodic mortality area in hectares
STATUS	rsrd hrvt avbl	reserved harvested non-reserved
SPECIES	*	species codes
SITE	*	site class codes
AGE	*	age class

PSCHART, SUNCHART, TEKCHART, IBMCHART

The PSCHART, SUNCHART, TEKCHART, and IBMCHART programs generate histograms from the summary data files. The PSCHART program generates a file containing "Postscript" page description commands. When printed on a "Postscript" printing device, the commands render the dataset as a bar chart. The SUNCHART command renders a histogram within a window on the screen of a SUN microcomputer. The TEKCHART program generates a sequence of graphics escape codes suitable for display on any Tektronics 4105 graphics terminal or emulator. IBMCHART is a MS-DOS program that displays images on an IBM-PC - compatible graphics screen. The syntax of the PSCHART, SUNCHART, and TEKCHART programs are as follows:

Syntax:
```
pschart <summary >psfile
sunchart <summary
tekchart <summary
ibmchart <summary
```

In the above example, `summary` would be replaced with the name of the input summary file, and `psfile` would be replaced with the name to be given to the output file. No other parameters are passed to the PS_CHART program. The output "Postscript" file may be printed on a Postscript laser printer using the UNIX `lpr` command.

TWOWAY

TWOWAY is a program that reformats a summary file into a two-way table suitable for processing with other programs. Files produced by TWOWAY may be loaded into database management systems, spreadsheet packages, or statistical programs. The syntax of the TWOWAY command is as follows:

Syntax:
```
twoway <summary >tabfile
```

In the above example `summary` would be replaced with the name of the input summary file and `tabfile` would be replaced with the name used to store the output table file.

TABLE

The table command reads in a two-way table produced by the TWOWAY command and produces a neatly formatted report suitable for output to a line printer. The report includes row and column headings, with subtotals printed at category breaks. The syntax of the table command is as follows:

Syntax:
```
table <tabfile >printfile
```

In the above example, `tabfile` would be replaced with the name of the table file produced by the TWOWAY program, and `printfile` would be replaced with a name to refer to the output print file.

Literature Cited

Baskerville, G. 1982. Use of WSM2 in forest dynamics labs. Department of Forestry, University of New Brunswick, Fredericton, N.B. Unpublished.

Hall, T.H. 1977. WOSFOP, instruction manual. New Brunswick Department of Natural Resources, Fredericton, N.B. Unpublished.

HENCO. 1985. Info reference manual. Henco Software Inc. Waltham, MA.

NBFRAC 1986. Yield curve construction for natural stands. N.B. Department of Natural Resources. Fredericton, N.B. Unpublished.

Newnham, M. 1988. FIRFOR - A simple forest management model. Forestry Canada Inf. Rep. PI-X-72.

Wang, E.; Erdle, T.; Rousell, T. 1987. Forman wood supply model user manual. New Brunswick Executive Forest Research Committee. Fredericton, N.B. Unpublished.

Acknowledgments

The authors gratefully acknowledge the time spent with the staff of Abitibi-Price Inc. and the Northern Region of Ontario Ministry of Natural Resources. Their efforts significantly improved the authors' understanding of problems facing forest managers in Northern Ontario, and made this a rewarding project to be involved in. In particular, the authors would like to thank Nick Salterelli and George Stanclik of Abitibi-Price Inc., Martin Litchfield and Grant Hauer of OMNR Northern Region, and Brian Callaghan of OMNR Sault Ste. Marie. This project was assisted by financial contributions from Abitibi-Price Inc. and OMNR Northern Region.

Appendix A

A sample run of the model

This sections shows a typical sequence of commands used to invoke the HSG model and display results. In this example, commands which would be typed into the computer are shown in a mono-spaced font. The particular syntax of commands shown in this example assumes a UNIX operating system.

1) Prepare the inventory datafile.

The inventory file contains one record for each productive forest stand being modelled. Typically, a program must be developed to convert inventory files in a "local" format to the "HSG" format. The inventory file contains data stored in binary format, thus the program that prepares the inventory file must be able to write binary fields. A program has been developed which converts the Ontario Ministry of Natural Resources "FRI stand file" to a file suitable for use with the HSG model. This program would be invoked with the following command

```
friter < fri.dat > stand. data
```

2) Prepare the eligibility database.

The eligibility database must be converted from "relational database" format to "HSG" format. A program is provided for this purpose. This program would be invoked with the following command

```
cnstprep < access.blk > access.cns
```

The above command suggests a filename extension of `.blk` for input relational database files, and `.cns` for the output eligibility database. Although any filename extension may be used, this consistency of application will assist in organizing and documenting datafiles.

3) Prepare the other datafiles.

The other datafiles used by the model must be prepared according to the format described in chapter 3. Most datafiles are in ASCII format and can be prepared using a system text editor.

4) Prepare the input parameter file.

A parameter file must be prepared which contains HSG control statements. The control statements identify required and optional datafiles, and provide rules and quotas to direct the simulation. The parameter file is given a filename extension of `.act` to indicate that it describes the activities that the model should simulate.

5) Invoke the model.

The model can be invoked under the control of an activity file with the following command:

```
hsg < basic.act > basic. sum
```

In the above, it is assumed that the activity file name is '`basic.act`'. The output file is given a filename extension of `.sum` to indicate that it contains summary information. When the model is invoked, a minimal amount of diagnostic information is displayed on the computer console, otherwise all results are stored in the `.sum` file or in files named in the HSG control statements.

6) Examine the contents of the summary file.

The summary file may be examined in several ways. In general, slices of information from the summary file may be extracted with the subset program. The data extracted by the subset program may be rendered with the chart programs to produce histograms or with the table program to produce formatted tables. The subset program requires a query string to identify the records to be selected. Chapter 5 describes the syntax of the query string in detail. The following examples produce a histogram and formatted table showing residual growing stock by species available over the course of the simulation.

```
subset -i 'YEAR:TYPE=volume:STATUS=avbl:SPECIES' < basic.sum | sunchart

subset -i 'TYPE=volume:STATUS=avbl:YEAR:SPECIES' < basic.sum | table
```

The above commands are formed as pipelines, where the output from one program feeds as input into the next. In this example the query strings are expressed differently for the table and chart program, although both request the same data be summarized. Changing the order of categories in the query string will affect the appearance of the table or chart. The table command generated n-way tables with the leftmost category in the query string as the outermost dimension in the table. In order to produce compact tables, categories with the greatest number of data codes should be specified on the right of the query string. The chart program generates histograms with the leftmost category in the query string along the x-axis of the chart.

7) Display the harvest schedule or snapshot files using the GIS.

If examination of the summary data indicates a satisfactory harvest level has been achieved, a detailed portrayal of the harvest schedule may be undertaken using the GIS. Although specifics of individual GIS systems are beyond the scope of this manual, the general process is to associate the files generated by the HSG model to the relational database system, to link the relational database files to the GIS, and to map out the harvest schedule using the choropleth mapping procedures of the GIS.

Appendix B. Program performance and compatibility

A major design goal in the development of these programs has been to insure portability between computer systems. All programs have been written in the "C" programming language for use on the SUN microcomputer. In addition, the programs were compiled to run under MSDOS on an IBM 80386 computer using the Pharlap virtual memory manager. It is anticiapted that these programs will run with minor modifications on any virtual memory computer that has a "C" compiler and the UNIX compatibility subroutines listed in Table B.1.

The software described in this document has been tested on a dataset for a 1.2 million hectare forest in the Iroquois Falls area of Northern Ontario. The inventory dataset consisted of 36 000 forest stands from a GIS database covering 161 inventory mapsheets digitized at a scale of 1:20 000. Results indicate that this model will complete ten 10-year simulation steps in 10 minutes on a SUN 4/110 microcomputer.

Program performance is highly dependent on the size of the forest to be analysed, and program memory usage increases in a linear manner with an increasing number of stands. The Iroquois Falls example required 8Mb of virtual memory for the 36 000 stand database.

Secondary storage (disk space) is required for the input and output datasets. The amount of disk space required depends on the size of the input datasets and the options used in the model. In general, snapshot files are the same size as the input inventory file, and the schedule file consumes 94 bytes for each output record. The size of the summary files will vary depending on the number of species, site class, and age class combinations in the forest and the number of periods over which the simulation was run. Simulation runs made using the Iroquois Falls dataset and 10 time steps yielded summary files that were 1 Mb in size.

Table B.1 Required system subroutines.

Routine name	Function
`alloc`, `calloc`	allocate virtual memory
`free`	release virtual memory
`atoi`	convert character to integer
`strcpy`, `strcmp`, `strtok`	character string utilities
`fscanf`, `sscanf`	formatted read
`fprintf`, `sprintf`	formated write
`fopen`	open system file
`fclose`	close system file
`fgets`, `getc`	read from file
`fputs`, `putc`	write data to file